

POST: enqueue

- Description: Enqueue the message to a Kafka topic.
- Example Fission setup:

```
fission route create --spec \  
    --name enqueue \  
    --url "/enqueue/{topic}" \  
    --method POST \  
    --createingress \  
    --function enqueue
```

- URL: /enqueue/{topic}
- Parameters:
 -

parameter	type	example	description
topic_type	string	"topic-mastodon"	the topic name

- Example usage:

```
requests.post(url=f'http://localhost:9090/enqueue/{topic_type}',  
              headers={'Content-Type': 'application/json'},  
              data=json.dumps({  
                  "index_name": index_name,  
                  "docs": json_data  
              }, default=str),  
              timeout=60)
```

- Return Body
 - status_code == 200: "success"
 - status_code == 500: "fail"

GET: search

- Description: Search relevant documents that contain keyword from Elasticsearch
- Example Fission setup:

```
fission route create --spec \
    --url '/search/{server:[a-zA-Z0-9]+}/keyword/{keyword:[a-zA-Z0-9]+}/start/{start:[0-9][0-9][0-9][0-9]-[0-1][0-9]-[0-3][0-9]}/end/{end:[0-9][0-9][0-9][0-9]-[0-1][0-9]-[0-3][0-9]}/size/{size:[0-9]+}' \
    --function search \
    --name search-get4 \
    --createingress \
    --method GET
```

- URL:
 - /search/{server}/keyword/{keyword}
 - /search/{server}/keyword/{keyword}/size/{size}
 - /search/{server}/keyword/{keyword}/start/{start}/end/{end}
 - /search/{server}/keyword/{keyword}/start/{start}/end/{end}/size/{size}

- Parameters:

-

parameter	type	example	description
server	string	"mastodon-en"	the Elasticsearch index name
keyword	string	"Tesla"	the keyword
start (optional)	string	"2022-04-18"	start date of docs (inclusive)
end (optional)	string	"2022-04-20"	end date of docs (not inclusive)
size (optional)	int	10	number of retrieved docs

- Example usage:

```
requests.get(url=f'http://localhost:9090/search/{server}/keyword/{keyword}',
             timeout=60)

requests.get(url=f'http://localhost:9090/search/{server}/keyword/{keyword}/size/{size}',
             timeout=60)

requests.get(url=f'http://localhost:9090/search/{server}/keyword/{keyword}/start/{start}/end/{end}',
             timeout=60)

requests.get(url=f'http://localhost:9090/search/{server}/keyword/{keyword}/start/{start}/end/{end}/size/{size}',
             timeout=60)
```

- Return Body
 - status_code == 200: list of retrieved docs
 - status_code == 500: "fail"

GET: retrieve

- Description: Retrieve all documents from Elasticsearch
- Fission setup:

```
fission route create --spec \
    --url '/retrieve/{server:[a-zA-Z0-9_]+}' \
    --function retrieve \
    --name retrieve-get1 \
    --createingress \
    --method GET
```

- URL: /retrieve/{server}
- Parameters:
 -

parameter	type	example	description
server	string	"incomepsnl"	the Elasticsearch index name

- Example usage:

```
requests.get(url=f'/retrieve/{server}',
             timeout=60)
```

- Return Body
 - status_code == 200: list of retrieved docs
 - status_code == 500: "fail"

GET: wordcloud

- Description: Return the word frequencies that can be used for plotting wordcloud
- Example Fission setup:

```
fission route create --spec \
    --url '/wordcloud/{server:[a-zA-Z0-9]+}/keyword/{keyword:[a-zA-Z0-9]+}/start/{start:[0-9][0-9][0-9][0-9]-[0-1][0-9]-[0-3][0-9]}/end/{end:[0-9][0-9][0-9][0-9]-[0-1][0-9]-[0-3][0-9]}/size/{size:[0-9]+}' \
    --function search \
    --name wordcloud-get4 \
    --createingress \
    --method GET
```

- URL:
 - /wordcloud/{server}/keyword/{keyword}
 - /wordcloud/{server}/keyword/{keyword}/size/{size}
 - /wordcloud/{server}/keyword/{keyword}/start/{start}/end/{end}
 - /wordcloud/{server}/keyword/{keyword}/start/{start}/end/{end}/size/{size}
- Parameters:
 -

parameter	type	example	description
server	string	"mastodon-en"	the Elasticsearch index name
keyword	string	"Tesla"	the keyword
start (optional)	string	"2022-04-18"	start date of docs (inclusive)
end (optional)	string	"2022-04-20"	end date of docs (not inclusive)
size (optional)	int	10	number of retrieved docs

- Example usage:

```
requests.get(url=f'http://localhost:9090/wordcloud/{server}/keyword/{keyword}',
             timeout=60)

requests.get(url=f'http://localhost:9090/wordcloud/{server}/keyword/{keyword}/size/{size}',
             timeout=60)

requests.get(url=f'http://localhost:9090/wordcloud/{server}/keyword/{keyword}/start/{start}/end/{end}',
             timeout=60)

requests.get(url=f'http://localhost:9090/wordcloud/{server}/keyword/{keyword}/start/{start}/end/{end}/size/{size}',
             timeout=60)
```

- Return Body
 - status_code == 200: word frequency dictionary that can be used for plotting wordcloud
 - status_code == 500: "fail"

